determined on a next system boot. This enables the full benefit of having the cache pre-warmed or fully occupied with data, because the user data and program code is already stored in the faster cache from previous user sessions. As a result, the system performance is improved on the next system boot/power on and data written to the cache but not yet written to the disk is preserved.

In the Claims

Following is a complete set of claims as amended with this response, which includes new claims 36-38.

1. A method comprising:

partitioning a non-volatile storage media;

storing data in a first partitioned section of the non-volatile storage media; and

storing, in a second partitioned section of the non-volatile storage media, metadata corresponding to the data stored in the first partitioned section of the non-volatile storage media.

2. The method of claim 1, wherein storing the metadata as packed metadata block.

3. The method of claim 1, wherein the partitioning is logical.

4. The method of claim 1, wherein storing cache data in the first partitioned section.

5. The method of claim 4, further comprising:

updating the data and metadata atomically when a line of cache data in the first partitioned section is changed.

6. The method of claim 1, further comprising:

allocating a portion of a mass storage device as the non-volatile storage media.

7. A non-volatile memory comprising:

a first section to store data; and

a second section partitioned from the first section, the second section to store metadata for the data stored in the first section.

8. The memory of claim 7, wherein the second section is to store the metadata as packed metadata blocks.

9. The memory of claim 7, wherein the partitioning of the first section and the second section is logical.

10. The memory of claim 7, wherein the non-volatile memory is a portion of a massive storage device.

11. The memory of claim 10, wherein the mass storage device is one of a disk drive, a Flash memory, a ferroelectric random access memory, or a polymer ferroelectric random access memory.

12. The memory of claim 7, wherein the non-volatile memory is a cache memory.

13. A system comprising:

a non-volatile storage media having a first section and a second section partitioned from the first section; and

a memory control hub to cause the first section to store data and the second section to store metadata for the data stored in the first section.

14. The system of claim 13, wherein second section is to store the metadata as packed metadata blocks.

15. The system of claim 13, wherein the partition is logical.

16. The system of claim 15, further comprising a massive storage device and wherein a portion of the massive storage device is the non-volatile storage media.

17. The system of claim 13, wherein the non-volatile storage media is a cache memory.

18. A method comprising:

partitioning a non-volatile storage media;

storing cache data in a first partitioned section of the non-volatile storage media;

storing metadata corresponding to the cache data in a second partitioned section of the non-volatile storage media; and

accessing the second partitioned section to determine the state of the cache data in a system boot.

19. The method of claim 18, wherein storing the metadata in the second partitioned section as packed metadata blocks.

20. The method of claim 18, wherein the partition is logical.

21. The method of claim 18, further comprising:

updating the cache data and metadata atomically when a line of cache data in the first partitioned section is changed.

22. A program loaded in a computer readable medium comprising:

a first group of computer instructions to logically partition a non-volatile storage media;

a second group of computer instructions to store data in a first partitioned section of the non-volatile storage media; and

a third group of computer instructions to store metadata for the data in a second partitioned section of the non-volatile storage media.

23. The program of claim 22, wherein the second group of computer instructions include computer instructions to store the metadata as packed metadata blocks.

24. The program of claim 22, wherein the second group of computer instructions include computer instructions to store cache data as the data in the first partitioned section.

25. The program of claim 24, further comprising:

computer instructions to update the data and metadata atomically when a line of cache data in the first partitioned section is changed.

26. The program of claim 24, further comprising:

computer instructions to access a line of the second partitioned section to read metadata for the cache data in the first partitioned section.

27. A program loaded in a computer readable medium comprising:

a first group of computer instructions to logically partition a non-volatile storage media;

a second group of computer instructions to store cache data in a first partitioned section of a non-volatile storage media;

a third group of computer instructions to store, in a second partitioned section of the non-volatile storage media, metadata corresponding to the cache data stored in the first partitioned section; and

a fourth group of instructions to access the second partitioned section to determine the state of the cache data.

28. The program of claim 27, wherein the third group of computer instructions includes computer instructions to store the metadata as packed metadata blocks.

29. The program of claim 27, further comprising:

computer instructions to update the cache data and metadata atomically when a line of cache data in the first partitioned section is changed.

30. The program of claim 27, further comprising:

computer instructions to allocate a portion of a mass storage device as the non-volatile storage media.

31. A system boot comprising:

accessing a first partitioned section of a non-volatile cache memory to read metadata for cache data stored in a second partitioned section of the non-volatile cache memory; and

determining the state of the cache data based upon the read metadata to initialize the non-volatile cache memory for the system boot.

32. The system boot of claim 31, wherein the metadata is stored in the first partitioned section as packed metadata blocks.

33. The system boot of claim 31, wherein the non-volatile cache memory is logically partitioned into the first and second partitioned sections.

34. The system boot of claim 31, further comprising: allocating a portion of a mass storage device as the non-volatile cache memory.

35. The system boot of claim 34, wherein the mass storage device is one of a disk drive, a Flash memory, a ferroelectric random access memory, or a polymer ferroelectric random access memory.

36. (New) A method comprising:

storing data in a non-volatile cache memory; and

storing metadata corresponding to the data stored in the non-volatile cache memory.

37. (New) The method of claim 36, further comprising:

updating the data and metadata atomically when a line of data in the non-volatile cache memory is changed.

38. (New) The method of claim 36, wherein the non-volatile cache memory is one of a disk drive, a Flash memory, a ferroelectric random access memory, or a polymer ferroelectric random access memory.